**What is Claimed is:**

1. A processor comprising:

a decoder to decode a load instruction naming a destination register;

a memory ordering buffer to maintain a source store instruction; and

a trailing store buffer to maintain an address for said source store instruction, if said source store instruction has been de-allocated from said memory ordering buffer.

2. The processor of claim 1 wherein said memory ordering buffer further comprises:

a store address buffer to maintain the address for said source store instruction.

3. The processor of claim 1 wherein said memory ordering buffer further comprises:

a store data buffer to maintain data associated with said source store instruction.

4. The processor of claim 1 further comprising:

a store data buffer coupled to said memory ordering buffer.

5. The processor of claim 1 wherein said trailing store buffer is coupled to said memory ordering buffer.

6. The processor of claim 1 wherein said memory ordering buffer comprises said trailing store buffer.

7. The processor of claim 1 further comprising a machine-readable medium having stored thereon a plurality of executable instructions to perform a method comprising:

computing a store address for said source store instruction;

writing the store address in a first storage;

writing data associated with the store address to a memory;

de-allocating the store address from the first storage;

allocating the store address in a second storage;

predicting a load instruction to be memory renamed;

computing a load store source index;

computing a load address;

disambiguating the memory renamed load instruction; and

retiring the memory renamed load instruction, if the load instruction reads its full data from a predicted store that is still allocated in at least one of said first storage and said second storage.

8. A method comprising:

computing a store address;

writing the store address in a first storage;

writing data associated with the store address to a memory;

de-allocating the store address from the first storage;

allocating the store address in a second storage;

predicting a load instruction to be memory renamed;

computing a load store source index;

computing a load address;

disambiguating the memory renamed load instruction; and

retiring the memory renamed load instruction, if the store address is still allocated in at least one of said first storage and said second storage.

9. The method of claim 8 wherein computing a store address comprises:

computing an address for a store instruction.

10. The method of claim 8 wherein writing the store address in a first storage comprises:

writing the store address in a store address buffer.

11. The method of claim 10 wherein writing data associated with the store address to a memory comprises:

writing the data from said store data buffer to said memory using the store address in said store address buffer.

12. The method of claim 11 wherein said store data buffer is in the first storage.

13. The method of claim 11 wherein said store data buffer is external to the first storage.

14. The method of claim 8 wherein de-allocating the store address from the first storage comprises:

de-allocating the store address from a store address buffer in the first storage.

15. The method of claim 8 wherein disambiguating the memory renamed load instruction comprises:

determining whether a source store address for the memory renamed load instruction corresponds to a store address in said first storage.

16. The method of claim 15 further comprises:

determining whether said source store address for the memory renamed load instruction is in the second storage.

17. The method of claim 8 further comprising:

clearing a backend of the processor and restarting the load instruction without memory renaming, if said source store address has been de-allocated from said first storage and said second storage.

18. A machine-readable medium having stored thereon a plurality of executable instructions to perform a method comprising:

computing a store address;

writing the store address in a first storage;

writing data associated with the store address to a memory;

de-allocating the store address from the first storage;

allocating the store address in a second storage;

predicting a load instruction to be memory renamed;

computing a load store source index;

computing a load address;

disambiguating the memory renamed load instruction; and

retiring the memory renamed load instruction, if the store address is still allocated in at least one of said first storage and said second storage.

19. The machine-readable medium of claim 18 wherein computing a store address comprises:

computing an address for a store instruction.

20. The machine-readable medium of claim 18 wherein writing the store address in a first storage comprises:

writing the store address in a store address buffer.

21. The machine-readable medium of claim 20 wherein writing data associated with the store address to a memory comprises:

writing the data from said store data buffer to said memory using the store address in said store address buffer.

22. The machine-readable medium of claim 21 wherein said store data buffer is in the first storage.

23. The machine-readable medium of claim 22 wherein said store data buffer is external to the first storage.

24. The machine-readable medium of claim 18 wherein deallocating the store address from the first storage comprises:

de-allocating the store address from a store address buffer in the first storage.

25. The method of claim 18 wherein disambiguating the memory renamed load instruction comprises:

determining whether a source store address for the memory renamed load instruction corresponds to a store address in said first storage.

26. The method of claim 25 further comprises:

determining whether said source store address for the memory renamed load instruction is in the second storage.

27. The method of claim 18 further comprising:

clearing a backend of the processor and restarting the load instruction without memory renaming, if said source store address has been de-allocated from said first storage and said second storage.

28. A computer system comprising:

a processor comprising:

a decoder to decode a load instruction naming a destination register;

a memory ordering buffer to maintain a source store instruction; and

a trailing store buffer to maintain an address for said source store instruction, if said source store instruction has been de-allocated from said memory ordering buffer; and

a memory coupled to said processor.

29. The computer system of claim 28 wherein said memory ordering buffer further comprises:

a store address buffer to maintain an address for said source store instruction.

30. The computer system of claim 28 wherein said memory ordering buffer further comprises:

a store data buffer to maintain data associated with said source store instruction.